



LAWRENCE
LIVERMORE
NATIONAL
LABORATORY

Adaptive Dynamic Bayesian Networks

B. M. Ng

October 30, 2007

2007 Joint Statistical Meetings
Salt Lake City, UT, United States
July 29, 2007 through August 2, 2007

Disclaimer

This document was prepared as an account of work sponsored by an agency of the United States government. Neither the United States government nor Lawrence Livermore National Security, LLC, nor any of their employees makes any warranty, expressed or implied, or assumes any legal liability or responsibility for the accuracy, completeness, or usefulness of any information, apparatus, product, or process disclosed, or represents that its use would not infringe privately owned rights. Reference herein to any specific commercial product, process, or service by trade name, trademark, manufacturer, or otherwise does not necessarily constitute or imply its endorsement, recommendation, or favoring by the United States government or Lawrence Livermore National Security, LLC. The views and opinions of authors expressed herein do not necessarily state or reflect those of the United States government or Lawrence Livermore National Security, LLC, and shall not be used for advertising or product endorsement purposes.

Adaptive Dynamic Bayesian Networks

Brenda Ng

Lawrence Livermore National Laboratory

Abstract

A discrete-time Markov process can be compactly modelled as a dynamic Bayesian network (DBN)—a graphical model with nodes representing random variables and directed edges indicating causality between variables. Each node has a probability distribution, conditional on the variables represented by the parent nodes. A DBN’s graphical structure encodes fixed conditional dependencies between variables. But in real-world systems, conditional dependencies between variables may be unknown a priori or may vary over time. Model errors can result if the DBN fails to capture all possible interactions between variables. Thus, we explore the representational framework of *adaptive* DBNs, whose structure and parameters can change from one time step to the next: a distribution’s parameters and its set of conditional variables are dynamic. This work builds on recent work in nonparametric Bayesian modeling, such as hierarchical Dirichlet processes, infinite-state hidden Markov networks and structured priors for Bayes net learning. In this paper, we will explain the motivation for our interest in adaptive DBNs, show how popular nonparametric methods are combined to formulate the foundations for adaptive DBNs, and present preliminary results.

KEY WORDS: Directed graphical models, dynamic Bayesian networks, nonparametric modeling

1. Introduction

Before one can perform any intelligent analysis for a given problem (e.g., fault diagnosis of a complex machine, prediction of disease spread, source inversion of environmental contaminants, etc.), one must first identify the quantities of interest and *model* these quantities in a coherent representation that supports efficient reasoning. Currently, many real-world processes are modelled by static models—models which are fixed throughout the lifetime of the process. This modeling paradigm is restrictive because static models often require an unrealistically abundance of prior knowledge about the process, such as:

- Number of random variables that can exist during the process lifetime
- Number of states that each variable can take on
- A complete enumeration of all possible interactions between variables

To address this issue, this work explores the approach of adaptive modeling—in particular, the design of adaptive models that can hopefully learn new features from data and revise

its representation accordingly. This capability will be especially useful for resource-bounded computation, since adaptive models can enable dynamic allocation of resources to represent only the features that are relevant to the reasoning task at hand. In contrast to static models, adaptive models relaxes the need for an a priori specification of all possible phenomena and interactions that may occur between the variables of interest. Thus, adaptive models can offer the most advantage for modeling dynamic processes whose dynamics are not well understood or may change over time, as in real-world applications of epidemiological modeling or adversarial modeling, where it is virtually impossible for any static model to capture all hypothetical scenarios that may occur between the entities of interest.

The take-home message of this work is that we need an **adaptive mechanism for modeling dynamic situations**. To address this need, we propose a new representational framework: *adaptive dynamic Bayesian networks*. Although this work is still in its nascent stage of development and verification, it is our hope that this paper may inspire synergistic efforts and collaborations between other researchers who may be interested in adaptive modeling.

The organization of this paper is as follows: We explain our motivation in Section 2 and review the technical background work in Section 3. Our technical contributions are detailed in Sections 4 and 5. We present preliminary results in Section 6 and conclude with ideas for future work in Section 7.

2. Motivation

Before we proceed with the technical discussion, we’d like to emphasize that our goal is to extend beyond model selection and our notion of adaptation is more than:

- the tweaking of parameters that scale the magnitude of a dynamic effect; or
- the selection of best scenario from a fixed set of prespecified scenarios.

Our long-term vision for this work is that adaptive models should facilitate:

- discovery of new hidden variables
- hypotheses of how these newly discovered hidden variables relate to known (hidden and observed) variables
- proposal of new representational features that may enhance the fidelity of the existing model

As a first step, we formulated a wishlist (cf. Table 1) of desirable properties for our adaptive model and used this wishlist

Table 1: Wishlist for adaptive models

- | |
|--|
| <ol style="list-style-type: none"> 1. Can accommodate an unbounded number of states 2. Can learn causal relationships between hidden variables 3. Can discover new hidden variables |
|--|

to guide our development. We began with dynamic Bayesian networks, a popular static modeling framework for temporal processes, and incrementally applied nonparametric modeling and Bayes net learning methodologies to morph these static networks into their adaptive counterparts. At the end, we achieved in addressing the first two items in this wishlist, while the last item is still work-in-progress.

3. Preliminaries

We start with explaining the notation that will be used for our technical discussion. Then, we will define dynamic Bayesian networks (DBNs), the foundation for this work. To transform a DBN into its adaptive variant, we will need two concepts: the nonparametric hidden Markov model and the structured prior. Thus, we will relate DBNs to hidden Markov models (HMMs), then illustrate how a HMM can be extended into its nonparametric counterpart—the hierarchical Dirichlet process HMM that has a countably infinite state space. Lastly, we review the concept of the structured prior and explain its role in Bayes net learning.

3.1 Notation

We use uppercase letters to denote random variables and lowercase letters to denote their instantiations. For example, given a binary variable $Z \in \{0, 1\}$, Z can be instantiated as either $z = 0$ or $z = 1$.

We use boldface when referring to a collection or set of similar items. For example, given two variables Z_1 and Z_2 , the collection of the two variables is referred to as $\mathbf{Z} = \{Z_1, Z_2\}$. Boldface is used for vectors as well.

Time will be indexed as a subscript. We use Z_t to denote a random variable Z at a specific time t , and $Z_{0:t}$ to denote the sequence of Z 's state from time 0 to time t . When referring to a particular variable Z in a collection of variables \mathbf{Z} at a given time t , if Z occurs as the n^{th} variable in \mathbf{Z} , then the said variable will be referred to as $Z_{n,t}$.

Lastly, we will be referring to the hidden state of a process at time t as \mathbf{S}_t . \mathbf{S}_t may be observed through noisy measurements, which in themselves are represented by observation variables \mathbf{Y}_t (that take on observed values \mathbf{y}_t).

3.2 Discrete-time Markov processes

A common approach to modeling temporal processes is to assume that processes evolve and are measured at equally spaced time steps. This is the key idea behind representing stochastic dynamic systems as discrete-time Markov processes.

In a discrete-time (first-order) Markov process, the current state captures all of the memory in the process, so that there is

no additional information in the past that can be used to predict the future. This (first-order) Markov property is expressed as:

$$p(\mathbf{S}_t | \mathbf{S}_{0:t-1}) = p(\mathbf{S}_t | \mathbf{S}_{t-1}), \quad t = 1, 2, \dots \quad (1)$$

In addition, observations depend only on the current state:

$$p(\mathbf{Y}_t | \mathbf{S}_{0:t}) = p(\mathbf{Y}_t | \mathbf{S}_t), \quad t = 1, 2, \dots \quad (2)$$

Thus, a discrete-time Markov process is characterized by two components: its transition model $p(\mathbf{S}_t | \mathbf{S}_{t-1})$ and its observation model $p(\mathbf{Y}_t | \mathbf{S}_t)$.

3.3 Dynamic Bayesian networks (DBNs)

Discrete-time Markov processes can be compactly represented by dynamic Bayesian networks (DBNs) [Dean and Kanazawa, 1989]. A DBN is the temporal version of a Bayesian network (cf. Figures 1 and 2). Like a Bayesian network, a DBN is a directed graphical model that represents a particular factorization of the joint distribution of all the variables in a given stochastic process. Each variable is represented as a node. A directed edge from node A to node B means that A influences B , or equivalently, A is a *parent* of B . Each node n is associated with a given conditional probability distribution $p(S_{n,t} | \mathbf{Pa}(S_{n,t}))$ that encapsulates the conditional probability of the variable $S_{n,t}$ given its parents $\mathbf{Pa}(S_{n,t})$. Given the values of its parents, a node is conditionally independent of its non-descendant nodes.

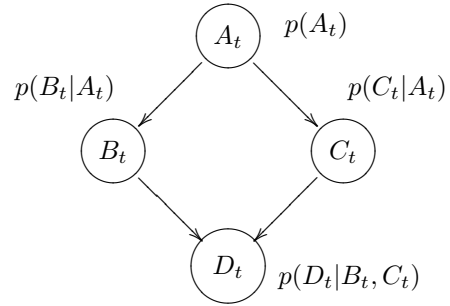


Figure 1: A Bayesian network

A DBN exploits the fact that, in most multivariable processes, each variable is typically influenced only by a local subset of the system variables. A DBN compactly represents the transition model $p(\mathbf{S}_t | \mathbf{S}_{t-1})$ in the factored manner imposed by its graphical structure:

$$p(\mathbf{S}_t | \mathbf{S}_{t-1}) = \prod_{n=1}^N p(S_{n,t} | \mathbf{Pa}(S_{n,t})) \quad (3)$$

where N is the number of variables in the state \mathbf{S}_t . The variables at each time step are assumed to be topologically sorted, such that $\mathbf{Pa}(S_{n,t}) \subseteq \{S_{1,t-1}, \dots, S_{N,t-1}\} \cup \{S_{1,t}, \dots, S_{n-1,t}\}$. In other words, a parent of the variable $S_{n,t}$ can be any variable from the previous state \mathbf{S}_{t-1} , or a variable in the current state \mathbf{S}_t that would not induce any cyclic dependencies.

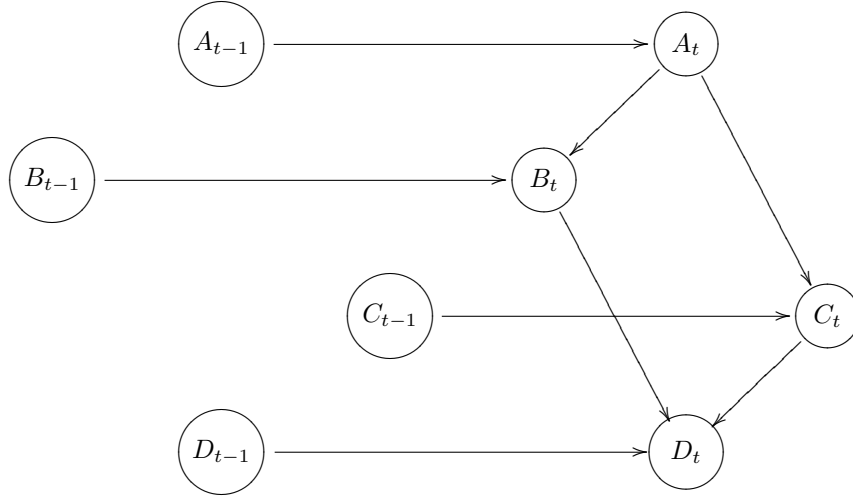


Figure 2: Extension of the Bayesian network in Figure 1 into a dynamic Bayesian network (DBN)

Specifically, a DBN is specified by a Bayesian network (that represents the probability distribution π_0 over the initial states) and a *2-time-slice Bayesian network* (that represents the transition distribution from states at time $t - 1$ to states at time t). For any terminal time T of a process, the joint distribution of its state from time 0 to T is given by:

$$p(\mathbf{S}_0 = \mathbf{s}_0, \mathbf{S}_1 = \mathbf{s}_1, \dots, \mathbf{S}_T = \mathbf{s}_T) = \pi_0(\mathbf{s}_0) \cdot \prod_{t=1}^T p(\mathbf{S}_t = \mathbf{s}_t | \mathbf{S}_{t-1} = \mathbf{s}_{t-1}) \quad (4)$$

3.4 Hidden Markov Models (HMMs)

A hidden Markov model (HMM) [Rabiner, 1989] can be represented as a simple DBN, where the state is represented by a single multinomial variable that can take on one of K discrete values, $S_t \in \{1, \dots, K\}$. For a time-invariant HMM, the transition model is specified by a K -by- K transition matrix (where each row defines a 1-by- K probability vector $P(S_{t+1}|S_t = s_t)$ for the current state s_t to transition to the next state). Similarly, if the observation is discrete-valued, such that $Y_t \in \{1, \dots, L\}$, then the observation model is specified by a K -by- L emission matrix (where each row defines a 1-by- L probability vector $P(Y_t|S_t = s_t)$).

3.5 Hierarchical Dirichlet Processes (HDPs)

We will be taking a nonparametric modeling approach in developing our adaptive models. A nonparametric model still has parameters, but is *nonparametric* in the sense that the set of possible models extends beyond a particular family of models that can be indexed by a finite number of parameters. Under this framework, it is assumed that the data are generated from a nonparametric model, with a potentially infinite number of parameters, but only a finite subset of these parameters is manifested in the actual data. Specifying a prior for these infinite dimensional parameters is tantamount to specifying a prior for random functions, which requires a stochastic process with realizations that are probability distributions.

One such suitable prior is the Dirichlet process (DP) [Ferguson, 1973; Antoniak, 1974]. A DP is a random probability measure on distributions. It is characterized by two parameters: a base distribution G_0 representing the center of the process and a precision parameter $\alpha_0 > 0$. A DP is a popular choice for a nonparametric prior on the parameters of a mixture model, because each draw $G \sim \text{DP}(\alpha_0, G_0)$ is (almost surely) a discrete distribution, represented as a countable mixture of point masses [Sethuraman, 1994]:

$$G = \sum_{j=1}^{\infty} \omega_j \delta_{\phi_j} \quad (5)$$

where

- $\{z_s\}_{s=1}^{\infty}$ and $\{\phi_j\}_{j=1}^{\infty}$ are independent sequences of i.i.d. random variables with $z_s \sim \text{Beta}(1, \alpha_0)$ and $\phi_j \sim G_0$;
- $\{\omega_j\}_{j=1}^{\infty}$ are the *stick-breaking weights*¹ defined as follows: $\omega_1 = z_1$ and $\omega_j = z_j \prod_{s=1}^{j-1} (1 - z_s)$.

In a situation where observations y_i arise from the distribution $K(\cdot; \theta_i)$, and θ_i is the parameter vector associated with the i^{th} component of a random mixture model:

$$\theta_i | G \sim G \quad (6)$$

$$y_i | \theta_i \sim K(\cdot; \theta_i) \quad (7)$$

if $G \sim \text{DP}(\alpha_0, G_0)$, then we have a DP mixture model:

$$F(\cdot; G) = \int K(\cdot; \theta) dG(\theta) \quad (8)$$

where $K(\cdot; \theta)$ can be any parametric family of distributions (e.g., Poisson, binomial, Gaussian, Gamma, etc.).

DP mixture models can be combined hierarchically to form *hierarchical Dirichlet process* mixture models [Teh *et al.*,

¹The procedure for generating $\omega \equiv \{\omega_j\}_{j=1}^{\infty}$ can be interpreted as iteratively breaking off the remaining portions of a unit-length stick. The weights ω defines a probability measure, since $\sum_{j=1}^{\infty} \omega_j = 1$ with probability one. The resulting *stick-breaking distribution* is written as $\omega \sim \text{Stick}(\alpha_0)$.

2006], for modeling groups of data where each group is characterized by a mixture model and mixture components are shared between groups. Intuitively, a hierarchical Dirichlet process (HDP) involves a set of DPs that are coupled via a common base measure, which in itself is also distributed according to a DP. Formally, imagine we have J disjoint groups of data and let y_{ji} be a single observation from the j^{th} group. In this context, Equations (6) and (7) can be extended to define a HDP:

$$G_0|\gamma, H \sim \text{DP}(\gamma, H) \quad (9)$$

$$G_j|\alpha_0, G_0 \sim \text{DP}(\alpha_0, G_0) \quad (10)$$

$$\theta_{ji}|G_j \sim G_j \quad (11)$$

$$y_{ji}|\theta_{ji} \sim K(\cdot; \theta_{ji}) \quad (12)$$

For each j , $\{\theta_{ji}\}_{i=1}^{\infty}$ represent the mixture components of the j^{th} group and are formulated as i.i.d. random variables distributed according to G_j . In turn, G_j (the group-specific base measure) stems from G_0 (the global base distribution), which is governed by H (the baseline measure).

3.6 Hierarchical Dirichlet Process HMMs

A HDP-HMM is the nonparametric variant of a HMM whose (multinomial) hidden variable can take on a countably infinite number of states. Recall from Subsection 3.4 that, in a traditional HMM, the number of values that S_t and Y_t can take on is finite and known a priori, thus it was possible to represent its transition model and its observation model by finite-dimensional matrices. If one interprets a state (a single value of the multinomial state variable) as a mixture component, then a HMM is essentially a *dynamic* finite mixture model, where each state corresponds to a mixture component, and the current state s_t serves as the indicator for which row of the transition matrix is to be used as the mixing proportions for choosing the next state. Thus, a HMM can be represented as a set of mixture models, one for each state.

Cast in the light of mixture models, the extension from a finite-state HMM to an infinite-state HMM is simple: It is achieved through replacing the state-conditional finite mixture models (that underlie the traditional HMM) with a HDP. The resulting model is a *hierarchical Dirichlet process hidden Markov model*² (HDP-HMM) [Teh *et al.*, 2006]. A HDP-HMM involves a set of DPs, one for each state. These DPs are coupled through a global DP, which enables the sharing of a common inventory of possible “next states” that can be reachable from each of the “current states”. Formally, each state k is associated with the transition parameters π_k and emission parameters ϕ_k , which fit into the HDP-HMM’s framework as follows:

$$\beta|\gamma \sim \text{Stick}(\gamma) \quad (13)$$

$$\pi_k|\alpha_0, \beta \sim \text{DP}(\alpha_0, \beta) \quad (14)$$

$$\phi_k|H \sim H \quad (15)$$

$$s_t|s_{t-1}, (\pi_k)_{k=1}^{\infty} \sim \pi_{s_{t-1}} \quad (16)$$

$$y_t|s_t, (\phi_k)_{k=1}^{\infty} \sim F(\cdot; \phi_{s_t}) \quad (17)$$

²Interested readers may want to refer to [Beal *et al.*, 2002] that describes a similar representation, the *infinite hidden Markov model*, which inspired the work of HDP-HMMs.

In Equation (13), the top-level state weights are sampled from the stick-breaking distribution (defined in Footnote 1). For each state $k \in \{1, 2, \dots\}$, the transition parameters π_k and emission parameters ϕ_k are drawn from the appropriate distributions defined in Equations (14) and (15). Then, for each time step $t \in \{1, \dots, T\}$, the state s_t and the observation y_t are generated according to the transition and emission parameters, as shown in Equations (16) and (17). A HDP-HMM can be interpreted as a HDP with a countably infinite number of groups. In contrast to the traditional HMM, the transition parameters π_k and emission parameters ϕ_k of a HDP-HMM are infinite-dimensional.

3.7 Structured priors

The theme of this paper is structural learning and adaptation for DBNs. Since a DBN is the temporal version of a Bayesian network, we examine the structural learning work for Bayesian networks to gain insights into DBN learning, of which we found the concept of *structured priors* [Mansinghka *et al.*, 2006] to be especially applicable.

In [Mansinghka *et al.*, 2006], a structured prior is a prior distribution over a set of candidate structures (i.e., directed acyclic graphs or DAGs) for a Bayesian network. Conditional on the structure, the parameters of the Bayesian network can be estimated through parameter estimation methods, as described in [Heckerman, 1999], therefore we defer the discussion on parameter learning and focus solely on the aspect of structural learning.

Assuming that variables (represented as nodes in a graph) belong to particular classes and these classes determine the prior probability that a directed edge exists between a node from one class and a node from a different class, one can specify the structured prior as having three parts:

- Partitioning of variables into classes: \mathbf{z} is a N -by-1 “class-assignment” vector that defines the partition of the N variables, whereby variables with the same class assignment have similar causal relationships. This partition is achieved through the Chinese Restaurant Process³ (CRP) [Pitman, 2002] with hyperparameter μ :

$$P(\mathbf{z}|\mu) = \mu^M \frac{\Gamma(\mu)}{\Gamma(N + \mu)} \prod_{m=1}^M (\psi_m - 1)! \quad (18)$$

where M is the number of existing classes and $\{\psi_m\}_{m=1}^M$ are the class weights, as defined in Footnote 3.

³The CRP draws an analogy between the partitioning of abstract objects (e.g., variables) and the random (table-sharing) seating arrangement in a Chinese restaurant. Imagine that a restaurant has a countably infinite number of tables, where each table corresponds to a class in a partition and the table assignment of the i^{th} customer is the class for the i^{th} object. Upon arrival, the i^{th} customer can choose to sit at any one of the M occupied tables with probability proportional to ψ_m , the number of customers already seated at table m ; or, sit at a new table with probability proportional to the hyperparameter μ . A CRP is closely related to a DP; the distinction is that a DP is a distribution over distributions, which induces a partitioning of variables, while a CRP is the corresponding distribution over partitions.

- Ordering of classes: \mathbf{o} is a M -by-1 vector, where the m^{th} element, o_m , represents the order of class m . This ordering imposes causal depth between the variables based on their class assignment. $P(\mathbf{o}|\mathbf{z})$ can be specified to encode a priori knowledge about the ordering or attributed with a uniform model (i.e., $P(\mathbf{o}|\mathbf{z}) = \frac{1}{M!}$).
- Probability of a directed edge from one class to another: $\boldsymbol{\eta}$ is a M -by- M matrix, where each element η_{o_a, o_b} is the probability of a directed edge from a node of class a to a node of class b . To avoid acyclic structures, only the strictly upper triangular entries, $\{\eta_{o_a, o_b}, o_b > o_a\}$, are non-zero and are distributed according to Beta(λ_1, λ_2):

$$P(\eta_{o_a, o_b}|\mathbf{z}) = \frac{\Gamma(\lambda_1)\Gamma(\lambda_2)}{\Gamma(\lambda_1 + \lambda_2)} \eta_{o_a, o_b}^{\lambda_1 - 1} (1 - \eta_{o_a, o_b})^{\lambda_2 - 1} \quad (19)$$

$\eta_{o_{z_i}, o_{z_j}}$ serves as the probability of whether an edge from variable i to variable j is present. An edge is treated as a random (Bernoulli) variable $\mathcal{G}_{i,j}$, where “ $\mathcal{G}_{i,j} = 1$ ” means an edge from i to j is present and “ $\mathcal{G}_{i,j} = 0$ ” means otherwise. In essence, $\mathcal{G} \equiv \{\mathcal{G}_{i,j}\}$ defines the adjacency matrix corresponding to a candidate graph for the unknown Bayesian network:

$$P(\mathcal{G}|\mathbf{z}, \mathbf{o}, \boldsymbol{\eta}) = \prod_{i=1}^N \prod_{j=1}^N \eta_{o_{z_i}, o_{z_j}}^{\mathcal{G}_{i,j}} (1 - \eta_{o_{z_i}, o_{z_j}})^{1 - \mathcal{G}_{i,j}} \quad (20)$$

Together, these three components define a probability distribution over possible structures for a Bayesian network. By applying Markov chain Monte Carlo inference, one can find approximations to the posterior for \mathbf{z} , \mathbf{o} and \mathcal{G} , and apply this knowledge to sample candidate structures that are supported by the observed data.

4. Hierarchical Dirichlet process DBNs (HDNs)

DBNs can be extended in the same spirit that a HMM is extended to a HDP-HMM, to formulate a nonparametric counterpart that can represent hidden variables, each with a possibly infinite number of states. In this framework, we assume that the number of hidden variables, along with the structure of the DBN, is fixed and known a priori. What is unknown is the number of states that each hidden variable can take on. Our proposal is a new model, the *hierarchical Dirichlet process dynamic Bayesian network* (HDN), that can learn the number of states for each variable from the observed data, thus addressing the first item on our wishlist (cf. Table 1).

Like a HMM, a DBN can be cast as a finite mixture model: each state of a hidden variable corresponds to a mixture component. However, the mixing proportions for choosing the next state of a hidden variable now depends not only on the current state of that variable, but on the current states of *all* its parent variables (which can include states from time t as well as time $t - 1$). In extending a finite-state DBN to an infinite-state DBN, we need to take care that this is properly expressed in the transition and emission parameters of our HDN framework. Let \mathcal{G} be the DAG that defines the DBN structure. From

\mathcal{G} , we can determine any variable’s set of parents: the current state of the n^{th} variable is denoted by $s_{n,t}$ and the states of its parent variables are denoted by the finite-dimensional vector $\boldsymbol{\rho}_{n,t}$. For each variable n , we have a separate HDP, whose parameters are indexed accordingly:

$$\boldsymbol{\beta}_n | \gamma_n \sim \text{Stick}(\gamma_n) \quad (21)$$

$$\boldsymbol{\pi}_{nk} | \alpha_{0n}, \boldsymbol{\beta}_n \sim \text{DP}(\alpha_{0n}, \boldsymbol{\beta}_n) \quad (22)$$

$$s_{n,t} | \boldsymbol{\rho}_{n,t}, (\boldsymbol{\pi}_{nk})_{k=1}^{\infty} \sim \boldsymbol{\pi}_n \cdot \boldsymbol{\rho}_{n,t} \quad (23)$$

If $s_{n,t}$ is observed through $y_{n,t}$, then additionally, we have:

$$\phi_{nk} | H_n \sim H_n \quad (24)$$

$$y_{n,t} | s_{n,t}, (\phi_{nk})_{k=1}^{\infty} \sim F_n(\cdot; \phi_{n, s_{n,t}}) \quad (25)$$

Adopting the plate notation⁴ [Jordan, 2004] to denote replication of subgraphs, we present the schematic for the HDN in Figure 3. In essence, a HDN is the multivariate version of the HDP-HMM, in the same way that a DBN generalizes a HMM from a univariate process. While a HDP-HMM can only represent univariate processes with a unbounded number of states, a HDN extends this capability for multivariate processes.

5. Structured priors for HDN learning

The development of HDN in Section 4 assumes that the structure \mathcal{G} is fixed and known a priori. If the structure is unknown, then \mathcal{G} is treated as a random variable and a probability distribution is placed on possible structures. As explained in Subsection 3.7, the structured prior offers a way for specifying such a distribution. In this section, we show how one can address the issue of unknown structures within the HDN framework by augmenting the HDN with the appropriate structured priors. This extension lifts the assumption of fixed/known structure previously imposed on HDNs and satisfies the second requirement on our wishlist (cf. Table 1).

Recall that a DBN is specified by two components:

- a Bayesian network \mathcal{B}_0 that defines the probability distribution over the states at an initial time step; structurally, \mathcal{B}_0 encodes only the *intra*-temporal structure: its edges originate from and end in the nodes from the same time slice;
- a 2-time-slice Bayesian network $\mathcal{B}_{\rightarrow}$ that defines the transition distribution from current states to next states; structurally, $\mathcal{B}_{\rightarrow}$ encodes the *inter*-temporal structure: edges originate from the previous time slice and end in the current time slice.

Analogously, we parallel our HDN learning process by decomposing the learning task into two parts:

- define a structured prior for the structure \mathcal{G}_0 that contains the intra-temporal edges

⁴The subgraph enclosed in a box or *plate* is replicated by the number of times indicated by the enclosed limit, located in the lower right corner of the plate. This notation provides a shorthand for representing diagrammatically repeated structures over many variables.

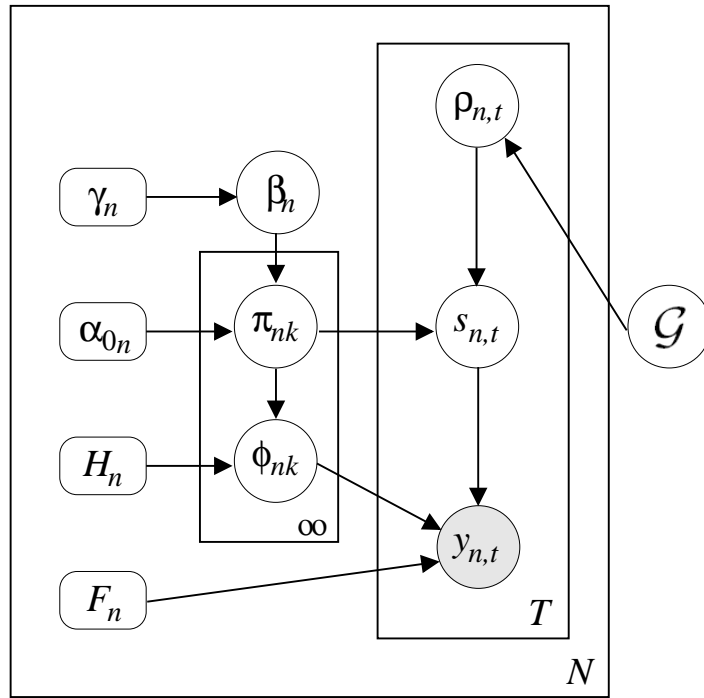


Figure 3: The hierarchical Dirichlet process DBN (HDN)

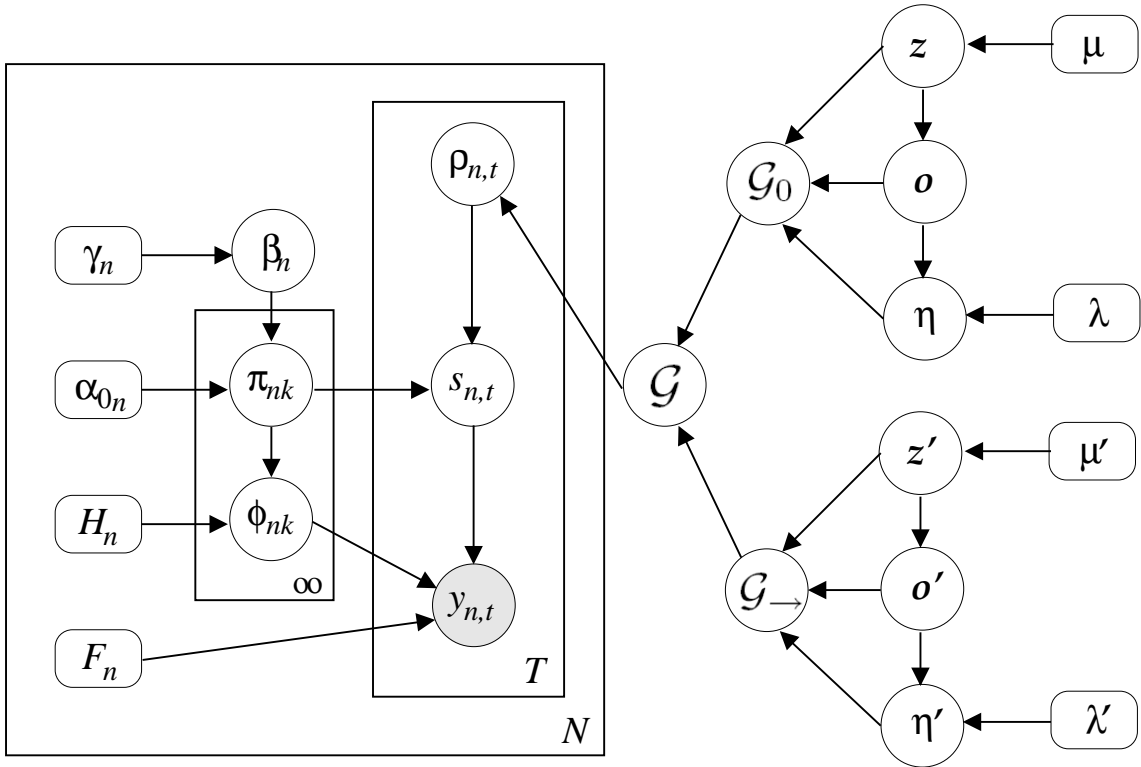


Figure 4: The HDN augmented with structure priors (HDN+)

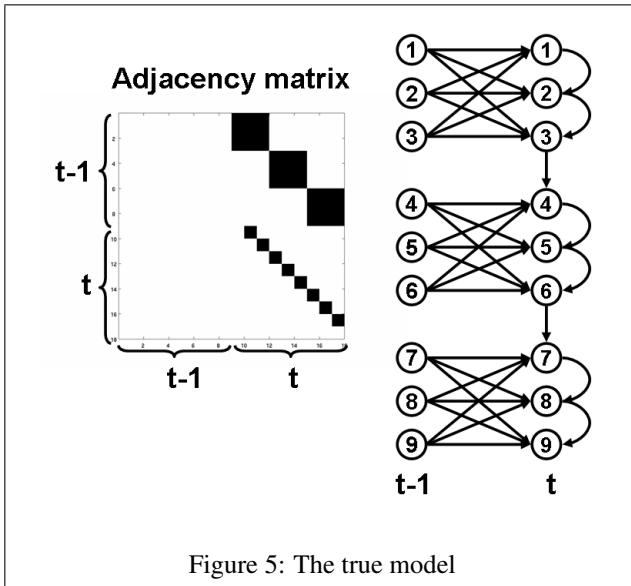


Figure 5: The true model

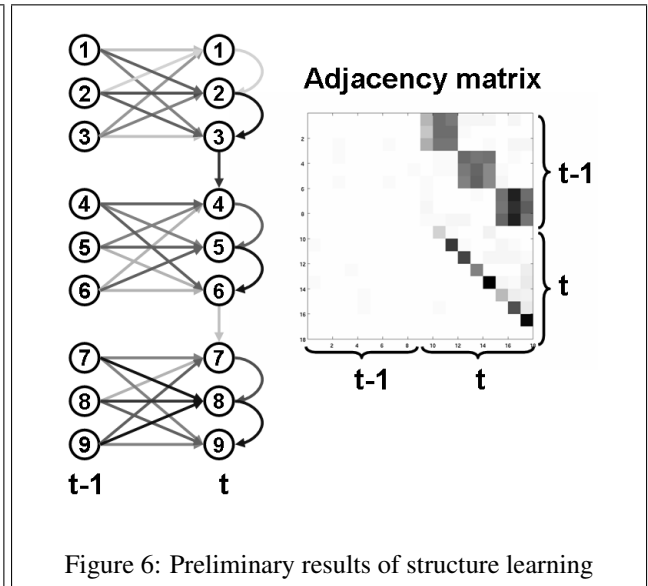


Figure 6: Preliminary results of structure learning

- define a structured prior for the structure $\mathcal{G}_{\rightarrow}$, that contains the inter-temporal edges

For this decomposition to work, the intra- and inter-temporal edges must be decoupled. This means that, for each node, the presence of any incoming intra-temporal edges must be independent of the presence of any inter-temporal edges (i.e., the existence of any particular intra-temporal edge should not affect the probability of another inter-temporal edge, in relation to the same node). After \mathcal{G}_0 and $\mathcal{G}_{\rightarrow}$ are inferred, the two are combined to form \mathcal{G} , the complete specification of the HDN's structure.

The schematic of the *augmented* HDN (denoted as HDN+) is presented in Figure 4. Note the two separate structured priors built on top of the HDN that allow for the representation of unknown structures within the framework of infinite-state models. With the appropriate choice of the hyperparameters governing the structured priors, it is our hope that we can extend this model to learning time-varying structures. For this to be achieved, we would probably be needing time-varying hyperparameters that can capture the evolution of the process structure. The design of these hyperparameters would be a difficult but interesting challenge.

6. Preliminary results

We apply the ideas outlined in Sections 4 and 5 to develop a HDN+ model for a simple multi-facility pipeline. The structure of the true model (that we used to generate the training data) is shown in Figure 5, where the adjacency matrix (left) and the DAG (right) encode the same information about the connectivity between the nodes. This model is an abstraction of three facilities: each facility is represented by three heavily coupled variables (in a clique formation) and the facilities pass materials or information from one to another in a sequential fashion.

We assumed that the number of variables (i.e., nodes in the network) are known and our goal is to infer the unknown structure using the time series data from the true model. Following

the Gibbs sampling procedures described in [Mansinghka *et al.*, 2006] and [Teh *et al.*, 2006], we first sample a candidate structure \mathcal{G} , then use \mathcal{G} 's specification of the parent sets to apply the auxiliary variable method of [Teh *et al.*, 2006] for inferring the HDP parameters of our model.

We present the results for the learned structures based on 100 trials of this inference procedure. The approximate posterior, constructed from the outcomes of these trials, is shown in Figure 6. In both the DAG (left) and the adjacency matrix (right), the intensity of an edge is proportional to the probability that the particular edge exists, in which edges with higher probabilities are shown as darker than those with lower probabilities. In the results, we see that all edges corresponding to those that are in the true model have non-zero probabilities; some edges are strongly predicted by the inferred posterior while others are only faintly manifested. In the near future, we hope to report more thoroughly on the learning performance, as a function of different amounts of data and/or different properties in the training network structures.

7. Conclusion and future work

In summary, this work was motivated by the philosophy that **when dealing with dynamic processes, models should evolve with the process**. Our passion for adaptive models stemmed from this idea. Clearly, adaptive models offer practical advantages over static models:

- Unbounded number of features: which allows for a flexible feature space that can adapt with the data
- No need to specify all possible phenomena a priori: which allows for a cleaner and more frugal representation that focuses on relevant features and interactions
- Modeling freedom: which lifts restrictive assumptions, such as constraints on parameters and/or structures, thus allowing the model to be more faithful to the actual process

In this paper, we proposed the framework of HDN+ as a starting point for developing adaptive DBNs. Future directions for extending this work, ranked in the order of increasing scope, include:

- Lifting the assumption on having a fixed number of hidden variables in the HDN (last item from the wishlist in Table 1), by augmenting the structural prior to allow for an infinite number of hidden causes, as explored in [Wood *et al.*, 2006]
- Incorporating time decay into the HDN framework such that temporal data from the distant past are more discounted than those from the recent past, by injecting time-sensitivity in the DP models [Zhu *et al.*, 2005]
- Improving the efficiency and scalability of inference and learning methods, such as exploring the feasibility of combining nonparametric modeling with optimization to enhance the adaptive models
- Inferring the time granularities for different parts of the process and incorporating this information into the resource management for scheduling inference and learning on the adaptive models, by drawing on insights from [Saria *et al.*, 2007]

Acknowledgments

This work was funded by the FY07 funding of the Predictive Knowledge Systems Strategic Initiative (LDRD project number 06-SI-006) and was performed under the auspices of the U.S. Department of Energy by Lawrence Livermore National Laboratory under Contract DE-AC52-07NA27344.

References

- [Antoniak, 1974] C. E. Antoniak. Mixtures of Dirichlet processes with applications to Bayesian nonparametric problems. *Annals of Statistics*, 2(6):1152–1174, 1974.
- [Beal *et al.*, 2002] M. J. Beal, Z. Ghahramani, and C. E. Rasmussen. The infinite hidden Markov model. In *Advances in Neural Information Processing Systems 14*, pages 577–584. MIT Press, 2002.
- [Dean and Kanazawa, 1989] T. Dean and K. Kanazawa. A model for reasoning about persistence and causation. *Computational Intelligence*, 5:142–150, 1989.
- [Ferguson, 1973] T. S. Ferguson. A Bayesian analysis of some non-parametric problems. *Annals of Statistics*, 1(2):209–230, 1973.
- [Heckerman, 1999] D. Heckerman. A tutorial on learning with Bayesian networks. In M. Jordan, editor, *Learning in Graphical Models*, pages 301–354. MIT Press, 1999.
- [Jordan, 2004] M. I. Jordan. Graphical models. *Statistical Science (Special Issue on Bayesian Statistics)*, 19:130–155, 2004.
- [Mansinghka *et al.*, 2006] V. K. Mansinghka, C. Kemp, J. B. Tenenbaum, and T. L. Griffiths. Structured priors for structure learning. In *Proceedings of the 22nd Annual Conference on Uncertainty in Artificial Intelligence (UAI 2006)*. AUAI Press, 2006.
- [Pitman, 2002] J. Pitman. Combinatorial stochastic processes. Technical Report 621, University of California at Berkeley, Department of Statistics, 2002. Lecture notes for École d’Été de Probabilités de Saint-Flour XXXII (2002).

- [Rabiner, 1989] L. R. Rabiner. A tutorial on hidden Markov models and selected applications in speech recognition. *Proceedings of the IEEE*, 77(2):257–286, 1989.
- [Saria *et al.*, 2007] S. Saria, U. Nodelman, and D. Koller. Reasoning at the right time granularity. In *Proceedings of the 23rd Annual Conference on Uncertainty in Artificial Intelligence (UAI 2007)*. AUAI Press, 2007.
- [Sethuraman, 1994] J. Sethuraman. A constructive definition of Dirichlet priors. *Statistica Sinica*, 4:639–650, 1994.
- [Teh *et al.*, 2006] Y. W. Teh, M. I. Jordan, and D. Blei. Hierarchical Dirichlet processes. *Journal of the American Statistical Association*, 101:1566–1581, 2006.
- [Wood *et al.*, 2006] F. Wood, T. L. Griffiths, and Z. Ghahramani. A non-parametric Bayesian method for inferring hidden causes. In *Proceedings of the 22nd Annual Conference on Uncertainty in Artificial Intelligence (UAI 2006)*. AUAI Press, 2006.
- [Zhu *et al.*, 2005] X. Zhu, Z. Ghahramani, and J. Lafferty. Time-sensitive Dirichlet process mixture models. Technical Report CMU-CALD-05-104, Carnegie Mellon University, School of Computer Science, 2005.